UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/075,871	02/13/2002	John C. Delo	2180.1	1152

7590	03/25/2005

Michalik & Wylie, PLLC
Suite 193
704-228th Avenue NE
Sammamish, WA 98074

EXAMINER
VO, TED T

ART UNIT	PAPER NUMBER
2192	

DATE MAILED: 03/25/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

	Application No.	Applicant(s)
Office Action Summary	10/075,871	DELO ET AL.
	Examiner	**Art Unit**	
	Ted T. Vo	2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address. --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 February 2002*.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-35* is/are pending in the application.

　　4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-35* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

　　Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

　　Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

　　a)☐ All　b)☐ Some * c)☐ None of:

　　　1.☐ Certified copies of the priority documents have been received.

　　　2.☐ Certified copies of the priority documents have been received in Application No. _____.

　　　3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
application from the International Bureau (PCT Rule 17.2(a)).

　　* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
　　Paper No(s)/Mail Date *2/11/03, 9/29/03*.

4) ☐ Interview Summary (PTO-413)
　　Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____ .

DETAILED ACTION

1.   This action is in response to the communication filed on 02/13/2002.

Claims 1-35 are pending in the application.

Information Disclosure Statement

2.   The references in the form PTO-1449 included in the information disclosure statement filed on

2/11/03, which are not marked with examiner initials, are not considered because they are not provided.

Double Patenting

3.   The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in
public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise
extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple
assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759
F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982);
In re Vogel, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and, *In re Thorington*, 418 F.2d 528, 163
USPQ 644 (CCPA 1969).
        A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome
an actual or provisional rejection based on a nonstatutory double patenting ground provided the
conflicting application or patent is shown to be commonly owned with this application. See 37
CFR 1.130(b).
        Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer.
A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

4.   Claims 1, 12, 21, 26, 30, 35 are rejected under the judicially created doctrine of obviousness-type

double patenting as being respectively unpatentable over claims 1, 9, 14, 16 of U. S. Patent No.

6,397,381 B1. Although the conflicting claims are not identical, they are not patentably distinct from each

other because:

As per Claim 1:

Claim 1 presented in this application and claims 1, 9, 16 in US 6,397,381 B1 are claiming the same

functionality.

As per Claim 12

Claim 12 presented in this application and claims 1, 9, 16 in US 6,397,381 B1 are claiming the same

functionality.

As per Claim 21

Claim 21 presented in this application and claim 9 in US 6,397,381 B1 are claiming the same

functionality.

As per Claim 26

Claim 26 presented in this application and claim 14 in US 6,397,381 B1 are claiming the same

functionality.

As per Claims 30, 35

Claims 30, 35 presented in this application and claims 1, 9, 16 in US 6,397,381 B1 are claiming the same

functionality.

Claim Rejections - 35 USC § 102

5.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for

the rejections under this section made in this Office action:

        A person shall be entitled to a patent unless –

        (a) the invention was known or used by others in this country, or patented or described in a printed
        publication in this or a foreign country, before the invention thereof by the applicant for a patent.

6.   Claim 1-35 are rejected under 35 U.S.C. 102(a) as being anticipated by Kelly, "Gain Control of

Application Setup and Maintenance with the New Windows Installer", Applicants' submitted prior art.

Given the broadest reasonable interpretation of followed claims in light of the specification.

As per Claim 1: Kelly discloses,

A computer-implemented method, comprising:

receiving a request to verify that a needed resource is available to an executable software program (See

section, 'How the Windows Installer Helps', particularly on Figure 1, where the Install service receive an

install transaction from 'user Privilege'), *the needed resource comprising at least one resource needed by*

the program (Figure 1, 'Install Service' pointing to registry and File system);

determining whether the needed resource is available to the program (Within the section, 'How the

Windows Installer Helps', refer to 'determining which software is installed'), *and if the needed resource is*

not available to the program, automatically initiating an installation procedure without manual termination

of the program to make the needed resource available to the program (See Figure 1, and within section

Installation Database, refer to 'INSTALLSTATE_ABSENT').

As per Claim 2: Kelly discloses, *The method of claim 1 wherein the resource comprises a registry key,*

and wherein determining whether the needed resource is available further comprises, querying a registry.

(See Figure 1, Install Service pointing to 'Registry', and Roolback Transaction).

As per Claim 3: Kelly discloses, *The method of claim 1 wherein the resource comprises a file, and*

wherein automatically initiating an installation procedure further comprises, installing the file at a storage

location accessible to the executable software program (See Figure 1, Install Service pointing to 'File*

System').

As per Claim 4: Kelly discloses, *The method of claim 1 wherein if the needed resource is available to the*

executable software program, returning existence verification data to the executable software program

(Within section Installation Database, and within paragraph INSTALLSTATE_ADVERTISED, refer to

'return to the path for an advertised component', also refer to Figure 4, 'Feature Installation states').

As per Claim 5: Kelly discloses, *The method of claim 4 wherein returning existence verification data comprises, returning a location of the resource to the executable software program* (Within section Installation Database, and within paragraph INSTALLSTATE_ADVERTISED, refer to 'return to the path for an advertised component').

As per Claim 6: Kelly discloses, *The method of claim 4 wherein returning existence verification data comprises, returning a resource path* (Within section Installation Database, and within paragraph INSTALLSTATE_ADVERTISED, refer to 'return to the path for an advertised component').

As per Claim 7: Kelly discloses, *The method of claim 1, wherein receiving a request to verify that a needed resource is available to an executable software program includes receiving a parameter identifying the resource* (Within section Installer Programming Interface, refer to 'application can identify itself to the installer', or 'Query the installer feature-by-feature').

As per Claim 8: Kelly discloses, *The method of claim 7, wherein determining whether the needed resource is available comprises, accessing a database based on the parameter identifying the resource to determine an expected location of the needed resource* (Within section Installation Database, refer to Installation States in Figure 4').

As per Claim 9: Kelly discloses, *The method of claim 8 wherein determining whether the needed resource is available to the executable software program further comprises, attempting to access the resource at the expected location* (Within section Installation Database, refer to Installation States in Figure 4' expected location such as'..._SOURCE': source installation media, '..._LOCAL': on local machine).

As per Claim 10: Kelly discloses, *The method of claim 1 wherein automatically initiating an installation procedure comprises, prompting the user to provide a source of the needed resource* (Within section Installation Database, refer to Installation States in Figure 4,'..._SOURCE' or '..._LOCAL', ..._ADVERTISED').

As per Claim 11: Kelly discloses, *A computer-readable medium having computer-executable instructions for performing the method of claim 1* (refer to the Installer Architecture of Figure 1).

As per Claim 12: Kelly discloses, *In a computing environment, a system comprising:*

an executable software program including a first set of executable code and a second set of executable

code, the second set of executable code comprising at least one resource that is needed to provide

functionality to the first set of executable code (Refer to INSTALLSTATE_ADVERTISED, and

INSTALLSTATE_ABSENT, as used by the installer in Figure , and/or/ the EXE or the DLL appeared

within section Installation Database); *and*

an installer program connected for communication with the first executable software program to receive a

request for installation information of the second set of executable code (Flow from Install Transaction to

Install Service), *the installer program configured to determine the installation information* (Flow from Install

Service to Registry, File System, Rollback Transaction), *and when the installation information indicates*

that the set of executable code is not installed (Within section Installation Database, refer to Figure 4,

"INSTALLSTATE_ABSENT", and refer Rollback Transaction operation in Figure 1), *the installer program*

further configured to automatically install the second set of executable code to make the second set of

executable code available to the first set of executable code (The operation perform by Figure 1,

according to 'Feature Installation States' given in Figure 4).

As per Claim 13: Kelly discloses, *The system of claim 12 wherein the first set of executable code*

comprises a product (i.e. INSTALLSTATE_ADVERTISED, or Install package given in Figure 1), *and*

wherein the second set of executable code corresponds to a feature of that product (See within

Installation Database).

As per Claim 14: Kelly discloses, *The system of claim 13 wherein the product includes a package file that*

describes at least one relationship between the feature and the at least one resource (See Figure 1 and

associated text of Install Package).

As per Claim 15: Kelly discloses, *The system of claim 12 wherein the first set of executable code*

comprises an application program, and wherein the second set of executable code comprises a

component including a collection of resources for that application program (See Figure 2 and associated

text within section Products, Feature, and Components).

As per Claim 16: Kelly discloses, *The system of claim 12 wherein the first set of executable code provides a token that includes data identifying the second set of executable code in the request for information received by the installer program* (See Figure 1, Registry within User privileges).

As per Claim 17: Kelly discloses, *The system of claim 12 wherein the token corresponds to a keypath, and wherein the installer program determines the installation information by checking for the existence of a file at a location based on the keypath* (Refer to Registry entry show within section Installation Database).

As per Claim 18: Kelly discloses, *The system of claim 12 wherein the second set of executable code comprises a component comprising a collection of resources, one of the resources comprises a key file, and wherein the first set of executable code provides data identifying the key file in the request for information received by the installer program* (See section, Products, Features, and Components).

As per Claim 19: Kelly discloses, *The system of claim 18 wherein the key file comprises a file system file, and wherein the installer program determines the installation information of the second set of executable code by looking for the existence of the key file at a storage location* (See section, Products, Features, and Components, and further see section Installation Database and descriptions in Figure 3).

As per Claim 20: Kelly discloses, *The system of claim 18 wherein the key file comprises registry data, and wherein the installer program determines the installation information based on the registry data* (See Figure 3 and descriptions of Registry and Remove Registry).

As per Claim 21: Kelly discloses, *A computer-readable medium having computer-executable instructions, comprising:*

receiving a resource identifier comprising at least one argument from a first set of executable code, the resource identifier being associated with a second set of executable code including at least one resource that provides functionality to the first set of executable code (See section Installer Programming interface);

accessing a database based on the resource identifier to retrieve an expected location of at least part of the second set of executable code; and verifying the existence of the at least part of the second set of

executable code at the expected location (See within section Installation Database, and within the

section, 'How the Windows Installer Helps', refer to 'determining which software is installed'), *and if*

verification is positive, passing the expected location to the first set of executable code (See within

section Installation Database, particularly see Figure 4) .

As per Claim 22: Kelly discloses, *The computer-readable medium of claim 21, further comprising,*

verifying the existence of the at least part of the second set of executable code at the expected location,

and if verification is negative, installing the second set of executable code at the expected location (See

within section Installation Database, particularly see Figure 4, 'INSTALLSTATE_ABSENT').

As per Claim 23: Kelly discloses, *The computer-readable medium of claim 21, wherein the resource*

identifier comprises a component code (e.g. 'path to the EXE or DLL' in section Installation Database)

which corresponding to at least one resource of the second set of executable code needed by the first set

of executable code (See within section Installation Database, particularly see Figure 4,

'INSTALLSTATE_ABSENT').

As per Claim 24: Kelly discloses, *The computer-readable medium of claim 21, wherein the resource*

identifier comprises a feature identifier which identifies at least one portion of the first set of executable

code (e.g. 'path to the EXE or DLL' in section Installation Database).

As per Claim 25: Kelly discloses, *The computer-readable medium of claim 21, wherein accessing the*

database to identify the expected location of the resource includes querying the database based on data

in the resource identifier (See whole section Installation Database).

As per Claim 26: Kelly discloses, *In a computing environment, a system comprising, executable code*

having a feature, the feature comprising a component including a key file which supports the feature; and

an installer for repairing the executable code if the key file becomes unavailable to the executable code,

the installer:

(a) receiving from the executable code a request for a path to the key file (Within section Installation

Database, "call the installer to get an actual path");

(b) identifying an expected location for the key file in the computing environment (Using identifiers such as LCID seen within section Qualified Component APIs, using Registry table within section Installation Database);

(c) attempting to verify the existence of the key file at the expected location in the computing environment (Within the section, 'How the Windows Installer Helps', refer to 'determining which software is installed', refer to Figure 4, using Feature Installation states); *and*

(d) in response to a failure to verify the existence of the key file at the expected location, automatically initiating an installation of the key file to the expected location without manual termination of the executable code (See Figure 1, Installer Architecture).

As per Claim 27: Kelly discloses, *The system of claim 26, wherein in response to successfully verifying the existence of the key file at the expected location, the installer provides data that identifies the expected location to the executable code* (Using identifiers such as LCID seen within section Qualified Component APIs, using Registry table within section Installation Database).

As per Claim 28: Kelly discloses, *The system of claim 26, wherein the key file corresponds to a resource of a component* (Refer to keys used in Registry, for example HKEY_CLASSES_ROOT).

As per Claim 29: Kelly discloses, *The system of claim 28, wherein the component corresponds to a feature* (Refer to keys used in Registry, for example HKEY_CLASSES_ROOT, and its path/subclass).

As per Claim 30: Kelly discloses, *In a computer system, a method comprising, receiving a call, the call including a resource identifier; and in response to receiving the call: 1) determining if a resource corresponding to the resource identifier exists at an expected location, and if the resource does not exist at the expected location, automatically initiating an installation of the resource to the expected location; and 2) returning information corresponding to the existence of the resource at the expected location.* Claim has the functionality corresponding to the limitation of Claim 1. See Rationale as addressed in Claim 1 above.

As per Claim 31: Kelly discloses, *The method of claim 30 wherein returning information corresponding to the existence of the resource at the expected location comprises returning a path to the expected location.* See Rationale as addressed in Claims 4-5 above.

<u>As per Claim 32</u>: Kelly discloses, *The method of claim 30 wherein determining if the resource corresponding to the resource identifier exists includes querying a database to obtain the expected location of the key file.* See Rationale as addressed in Claim 2 above.

<u>As per Claim 33</u>: Kelly discloses, *The method of claim 32, wherein if the expected location cannot be found by querying the database, automatically initiating an installation of the resource to a location and adding that location to the database as the expected location.* See Rationale as addressed in Claim 2 above.

<u>As per Claim 34</u>: Kelly discloses, *The method of claim 30 wherein the resource identifier corresponds to a key file, and wherein determining if a resource corresponding to the resource identifier exists at an expected location comprises obtaining a path to that key file* (See within section Installation Database, "call the installer to get an actual path. Using identifiers such as LCID seen within section Qualified Component APIs, using Registry table within section Installation Database).

<u>As per Claim 35</u>: Kelly discloses, *A computer-readable medium having computer-executable instructions for performing the method of claim 30.* See Rationale as addressed in Claim 1 above.

Conclusion

7.   The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.
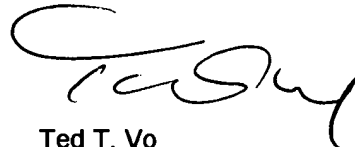
Gottfried Rudorfer, "Managing PC Operating Systems with a Revision Control System", discloses installation and repair Linux software.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ted T. Vo whose telephone number is (571) 272-3706. The examiner can normally be reached on 8:00AM to 5:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3694. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Ted T. Vo
Patent Examiner
Art Unit 2122
March 18, 2005